Amendments to the Claims

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (original) In a data-packet processor, a system for non-speculative pre-fetching, comprising:

   a processing unit having a first portion for processing the data packets, using instruction threads specific to packet type, and a second portion comprising a pool of context registers and functional units for processing;

   a packet-management unit (PMU) for managing data packets of 10 different types received for processing, including selecting and loading the context registers;

   a memory storing at least an initial instruction of instruction threads; and

   a table equating packet types with pointers to memory locations for the at least first instructions of instruction threads specific to the packet types;

   characterized in that the PMU selects a context from the pool of contexts for processing of a data packet, the table is consulted for the pointer, and the pointer is provided to the processing unit first portion, enabling the processing unit first portion to prefetch at least an initial instruction for the packet to be processed at least partially in parallel with loading of the context.

2. (original) The system of claim 1 wherein the second portion of the processing unit comprises separate clusters, each cluster comprising contexts and functional units.

3. (original) The system of claim 1 wherein the table is in the PMU.

4. (original) The system of claim 1 wherein the processor is a dynamic multi-streaming processor.

5. (original) The system of claim 1 wherein the memory holding at least a first instruction of the instruction threads is an on-chip instruction cache memory.

6. (original) The system of claim 1 wherein the memory holding at least a first instruction of the instruction threads is an off-chip memory.

7. (original) The system of claim 1 wherein data packets to be processed are stored in queues according to instruction threads required to process the packets, and wherein the queue from which a packet arrives for processing indicates the packet type.

8. (original) In a data-packet processor having a first portion for processing data packets, using instruction threads specific to packet type, and a second portion comprising a pool of context registers and functional units for processing, a method for accomplishing pre-fetch of at least a first 20 instruction for processing, comprising steps of:

(a) selecting, by a packet-management unit (PMU), an available context for loading information for processing a packet ready for processing;

(b) consulting a table relating packet type for the packet ready to be processed to a pointer to a memory location for at least a first instruction of an instruction thread to process the packet;

(c) providing the pointer to the first portion; and

(d) pre-fetching the at least first instruction of the thread to process the data packet, at least partially in parallel with loading the context.

9. (currently amended) The method of claim 8 wherein the second portion ~~of the processing unit~~ comprises separate clusters, each cluster comprising contexts and functional units.

10. (original) The method of claim 8 wherein the table is in the PMU.

11. (original) The method of claim 8 wherein the processor is a dynamic multi-streaming processor.

12. (original) The method of claim 8 wherein the memory holding at least a first instruction of the instruction threads is an on-chip instruction cache memory.

13. (original) The method of claim 8 wherein the memory holding at least a first instruction of the instruction threads is an off-chip memory.

14. (original) The method of claim 8 wherein data packets to be processed are stored in queues according to instruction threads required to process the packets, 20 and wherein the queue from which a packet arrives for processing indicates the packet type.